Numerical implementation and oceanographic application of the Gibbs potential of ice

R. Feistel, W. Wagner, V. Tchijov, C. Guder

Ocean Science, 1, 29–38, 2005
www.ocean-science.net/os/1/29/
SRef-ID: 1812-0792/os/2005-1-29
European Geosciences Union

Ocean Science

Numerical implementation and oceanographic application of the Gibbs potential of ice

R. Feistel[1], **W. Wagner**[2], **V. Tchijov**[3], **and C. Guder**[2]

[1]Baltic Sea Research Institute, Seestraße 15, 18119 Warnemünde, Germany
[2]Ruhr-Universität Bochum, Lehrstuhl für Thermodynamik, 44780 Bochum, Germany
[3]Centro de Investigaciones Teóricas, Facultad de Estudios Superiores Cuautitlán, UNAM, México

Received: 21 December 2004 – Published in Ocean Science Discussions: 3 February 2005
Revised: 29 March 2005 – Accepted: 14 April 2005 – Published: 31 May 2005

Abstract. The 2004 Gibbs thermodynamic potential function of naturally abundant water ice is based on much more experimental data than its predecessors, is therefore significantly more accurate and reliable, and for the first time describes the entire temperature and pressure range of existence of this ice phase. It is expressed in the ITS-90 temperature scale and is consistent with the current scientific pure water standard, IAPWS-95, and the 2003 Gibbs potential of seawater. The combination of these formulations provides sublimation pressures, freezing points, and sea ice properties covering the parameter ranges of oceanographic interest. This paper provides source code examples in Visual Basic, Fortran and C++ for the computation of the Gibbs function of ice and its partial derivatives. It reports the most important related thermodynamic equations for ice and sea ice properties.

1 Introduction

Oceanography demands very accurate knowledge of freezing temperatures of seawater. For instance, global temperature rise, as manifested in the 1.2°C increase per century of northern hemisphere land surface winter temperatures (Jones, 1994; Hagen and Feistel, 2005), is in general accompanied by much smaller changes of ocean temperatures. But by freezing and melting processes in the polar and mid latitudes, even a tiny cooling or warming of the sea can nonetheless have significant, nonlinear impacts on the climate system, e.g. in form of growing or shrinking sea ice covers.

Several freezing point formulae are available for water and seawater. The Unesco fomula of Millero (1978) is valid for air-saturated seawater at low pressures only (up to 5 MPa=500 dBar) and does not take into account compressibilities of seawater and ice. In the zero salinity limit, it is not consistent with the international equation for the melting pressure of Wagner et al. (1994) for air-free water which covers more than the oceanic pressure range up to 100 MPa. Neither formula, moreover, does agree with the most accurate measured Clausius-Clapeyron coefficient within its experimental uncertainty (Feistel and Wagner, 2005b).

Thermodynamic potential functions (also called fundamental or general equations of state) offer a very compact and consistent way of representing equilibrium properties of a given substance, both theoretically and numerically (Alberty, 2001). This was very successfully demonstrated by subsequent standard formulations for water and steam (Wagner and Pruß, 2002) and other fluid substances (Span and Wagner, 2003). With the numerical availability of mutually consistent formulations for the chemical potentials of water (Wagner and Pruß, 2002) and seawater (Feistel, 2003), a Gibbs function approach to the freezing point computation problem became appealing. Earlier Gibbs functions of ice (Feistel, 1993; Feistel and Hagen, 1995; Tillner-Roth, 1998; Feistel, 2003) were based on only few experimental data from the vicinity of the normal pressure melting point, and suffered from e.g. the pending significant uncertainty of ice compressibility. Therefore, a reliable and accurate Gibbs function of ice was desired, consistent with the corresponding latest descriptions of water, vapour, and seawater, which can provide freezing points of water and seawater over the entire oceanographic "Neptunian" pressure range. An explicit freezing point formula derived from this newly developed Gibbs formulation is given in Jackett et al. (2005).

When water or seawater freezes under natural conditions, crystals of hexagonal, so-called ice Ih are formed. At high pressures, ice possesses at least twelve other crystalline phases and two amorphous states, exclusively discovered in the laboratory. In the current formulation, only the naturally abundant ice phase, ice Ih, is described, applicable to the entire region of its existence from 0 to 210 MPa in pressure und from 0 to 273.16 K in temperature.

Correspondence to: R. Feistel
(rainer.feistel@io-warnemuende.de)

Table 1. Special constants and values used in this paper.

quantity	symbol	value	unit	abs. error
triple point pressure	P_t	611.657	Pa	0.010
normal pressure	P_0	101325	Pa	exact
triple point temperature	T_t	273.160	K	exact
Celsius zero point	T_0	273.150	K	exact
normal melting point	T_M	273.152518	K	2E-6
temperature scale factor	$\tau = T_t/1\ \mathrm{K}$	273.160	–	
pressure scale factor	$\psi = P_t/1\ \mathrm{Pa}$	611.657	–	
normal pressure constant	$p_0 = P_0/P_t$	165.6565689594	–	
unit specific free enthalpy	g_U	1	J/kg	
unit specific entropy	σ_U	1	J/(kg K)	
unit specific volume	v_U	1	m^3/kg	
unit specific heat capacity	c_U	1	J/(kg K)	

A detailed derivation of this function is given by Feistel and Wagner (2005a, b). It was determined by regression with respect to 339 data points belonging to 26 different groups of experiments, reproducing these data within their particular measurement uncertainties. The new formulation obeys Debye's cubic law at low temperatures and pressure-independence of the residual entropy. The typical uncertainty of about 100% in isothermal compressibilities of previous formulae is reduced to less than 1%. The resulting Clausius-Clapeyron slope of the melting curve at normal pressure is 74.305 mK/MPa with an uncertainty of only 0.02%, which is about 100 times more accurate than the corresponding figures of the formulae of Millero (1978) or Wagner et al. (1994). The fresh water melting temperature at normal pressure is inferred to be 273.152 518±0.000 002 K. Recomputed seawater freezing points are given in Table 7; their accuracy is estimated as 2 mK at atmospheric pressure and 30 mK at high pressures (Feistel and Wagner, 2005b).

A major advantage of the Gibbs function method applied to sea ice is that it suffices to study the freezing formula for pure water, which then in turn automatically implies the formula for seawater, too. The chemical potential of ice, once available, predicts the freezing curve of water if compared with the chemical potential of water, available from the IAPWS-95 formulation (Wagner and Pruß, 2002). Compared with the chemical potential of water in seawater, available from the Gibbs function of seawater (Feistel, 2003), it as well provides the freezing curve of seawater with comparable accuracy.

The current paper provides numerical implementations of the new Gibbs function of ice and its derivatives in Visual Basic (VB), Fortran, and C++. These implementations have been developed independently by the different authors of this paper, and are subject to specific properties of the particular programming language and coding rules. To give an example, VB does not support complex arithmetic and data types, which needed to be implemented explicitly. Thus, the three source code versions differ from each other in various details, and are therefore described separately one by one in Sect. 3. These code examples are intended as functioning examples and possible guides for the development of individual implementations into custom program environments. They do not provide any user interface nor support data input or graphical output.

For reference, a collection of explicit expressions for the partial derivatives of the Gibbs function, as well as various relations to other thermodynamic properties are given in Sect. 2. Finally, Sect. 4 presents formulae for sea ice, which additionally require for their evaluation the Gibbs potential of seawater. Numerical implementations of the 2003 Gibbs function of seawater in VB, Fortran, and C++ are available from a companion article earlier in this journal (Feistel, 2004, 2005).

2 Gibbs potential and its derivatives

The thermodynamic Gibbs potential function $g(T, P)$ described in this paper is the specific free enthalpy (specific Gibbs energy) of ice, which is equal to the chemical potential $\mu(T, P)$ of ice. We express absolute temperature T by the dimensionless variable $t = T/T_t$ with triple point temperature T_t, and absolute pressure P by $p = P/P_t$ with triple point pressure P_t. Triple point data and other constants are reported in Table 1.

The functional form of $g(T, P)$ for ice is given as a complex function of temperature,

$$\frac{g(T,P)}{g_U} = g_0 - \sigma_0\tau\cdot t + \tau\,Re\left\{\sum_{k=1}^{2} r_k\left[(t_k-t)\ln(t_k-t) + (t_k+t)\ln(t_k+t) - 2t_k\ln t_k - \frac{t^2}{t_k}\right]\right\}, \quad (1)$$

with two coefficients being polynomials of pressure,

$$g_0(P) = \sum_{k=0}^{4} g_{0k} \cdot (p - p_0)^k \,, \; r_2(P) = \sum_{k=0}^{2} r_{2k} \cdot (p - p_0)^k \,. \quad (2)$$

The unit specific free enthalpy is $g_U = 1$ J/kg, the dimensionless constant τ is $\tau = T_t/1\mathrm{K} = 273.160$ and the dimensionless normal pressure is $p_0 = P_0/P_t$, given in Table 1. The real constants g_{00} to g_{04} and σ_0 as well as the complex constants t_1, r_1, t_2 and r_{20} to r_{22} are given in Table 2.

The complex logarithm $\ln(z)$ is the principal value, i.e. it evaluates to imaginary parts in the interval $-\pi < Im\ \ln(z) \leq +\pi$. The complex notation used here serves only for compact writing and easy determination of the partial derivatives. The residual entropy coefficient σ_0 given in Table 2 is its "IAPWS-95 version" as required for correct phase equilibria determinations between ice and pure water using the IAPWS-95 formulation (Wagner and Pruß, 2002), or seawater (Feistel, 2003). Its alternative value represents the true absolute physical zero-point entropy of ice, $\sigma_0 = 189.13$, which is usually not required for practical computations, however.

The first derivative of g with respect to its independent variable P, provides density, ρ, and specific volume, v,

$$\frac{1}{\rho} = v = \left(\frac{\partial g}{\partial P}\right)_T$$

$$\frac{v}{v_U} = \frac{1}{\psi}\frac{\mathrm{d}g_0}{\mathrm{d}p} + \frac{\tau}{\psi} Re \left\{ \frac{\mathrm{d}r_2}{\mathrm{d}p} \left[(t_2 - t)\ln(t_2 - t) + (t_2 + t)\ln(t_2 + t) - 2t_2 \ln t_2 - \frac{t^2}{t_2} \right] \right\}$$

$$\psi = \frac{P_t}{1\,\mathrm{Pa}} = 611.657,$$

$$\frac{\mathrm{d}g_0}{\mathrm{d}p} = \sum_{k=1}^{4} g_{0k} \cdot k \cdot (p - p_0)^{k-1}, \; \frac{\mathrm{d}r_2}{\mathrm{d}p} = \sum_{k=1}^{2} r_{2k} \cdot k \cdot (p - p_0)^{k-1} \quad (3)$$

and the one with respect to T provides specific entropy, σ,

$$\sigma = -\left(\frac{\partial g}{\partial T}\right)_P \quad (4)$$

$$\frac{\sigma}{\sigma_U} = \sigma_0 + Re \left\{ \sum_{k=1}^{2} r_k \left[\ln\left(\frac{t_k - t}{t_k + t}\right) + 2\frac{t}{t_k} \right] \right\}$$

Further thermodynamic functions are defined by so-called Legendre transforms:

Specific free energy (also called Helmholtz energy or Helmholtz free energy), f,

$$f = g - Pv = g - P \cdot \left(\frac{\partial g}{\partial P}\right)_T \quad (5)$$

Specific enthalpy, h,

$$h = g + T\sigma = g - T \cdot \left(\frac{\partial g}{\partial T}\right)_P \quad (6)$$

Table 2. Coefficients of the Gibbs functions (Eqs. 1, 2).

Coefficient	Real part	Imaginary part
g_{00}	−632578.704355102	
g_{01}	0.655029997804786	
g_{02}	−1.89952376891314E-08	
g_{03}	3.40692612753936E-15	
g_{04}	−5.78593658679522E-22	
σ_0	−3333.18160308627	
t_1	3.71539090346389E-02	5.10464771184122E-02
r_1	45.951447199735	65.223705014775
t_2	0.345095829562823	0.343315892017841
r_{20}	−75.8695106343435	−80.9878506462645
r_{21}	−5.75529765634353E-05	5.09059011946526E-05
r_{22}	2.39617513518116E-11	−2.73297877749166E-11

Specific internal energy, e,

$$e = g + T\sigma - Pv = g - T \cdot \left(\frac{\partial g}{\partial T}\right)_P - P \cdot \left(\frac{\partial g}{\partial P}\right)_T \quad (7)$$

Several thermodynamic coefficients can be derived from second derivatives of g:

Isothermal compressibility, K,

$$K = -\frac{1}{v}\left(\frac{\partial v}{\partial P}\right)_T = -\frac{\left(\partial^2 g/\partial P^2\right)_T}{(\partial g/\partial P)_T} \quad (8)$$

$$\frac{P_t}{v_U \psi}\left(\frac{\partial v}{\partial P}\right)_T = \frac{1}{\psi^2}\frac{\mathrm{d}^2 g_0}{\mathrm{d}p^2} + \frac{\tau}{\psi^2} Re \left\{ \frac{\mathrm{d}^2 r_2}{\mathrm{d}p^2} \left[(t_2 - t)\ln(t_2 - t) + (t_2 + t)\ln(t_2 + t) - 2t_2 \ln t_2 - \frac{t^2}{t_2} \right] \right\}$$

$$\frac{\mathrm{d}^2 g_0}{\mathrm{d}p^2} = \sum_{k=2}^{4} g_{0k} \cdot k\,(k-1) \cdot (p - p_0)^{k-2}, \quad \frac{\mathrm{d}^2 r_2}{\mathrm{d}p^2} = 2r_{22}$$

Isobaric cubic thermal expansion coefficient, α,

$$\alpha = \frac{1}{v}\left(\frac{\partial v}{\partial T}\right)_P = \frac{\left(\partial^2 g/\partial T \partial P\right)}{(\partial g/\partial P)_T} \quad (9)$$

$$\frac{T_t}{v_U \tau}\left(\frac{\partial v}{\partial T}\right)_P = \frac{1}{\psi} Re \left\{ \frac{\mathrm{d}r_2}{\mathrm{d}p} \left[\ln\left(\frac{t_2 + t}{t_2 - t}\right) - 2\frac{t}{t_2} \right] \right\}$$

Specific isobaric heat capacity, c_P,

$$c_P = T\left(\frac{\partial \sigma}{\partial T}\right)_P = \left(\frac{\partial h}{\partial T}\right)_P = -T\left(\frac{\partial^2 g}{\partial T^2}\right)_P \quad (10)$$

$$\frac{c_P}{c_U} = t \cdot Re \left\{ \sum_{k=1}^{2} r_k \left[\frac{1}{t - t_k} - \frac{1}{t + t_k} + \frac{2}{t_k} \right] \right\}$$

Table 3. Factorisation of the Gibbs potential function, Eq. (13).

k	$\Theta_k(T)$	$\Phi_k(P)$
1	$\tau\cdot\left[(t_1-t)\ln(t_1-t)+(t_1+t)\ln(t_1+t)-2t_1\ln t_1-\frac{t^2}{t_1}\right]$	r_1
2	$\tau\cdot\left[(t_2-t)\ln(t_2-t)+(t_2+t)\ln(t_2+t)-2t_2\ln t_2-\frac{t^2}{t_2}\right]$	$\sum_{j=0}^{2} r_{2j}\cdot(p-p_0)^j$
3	$\tau\cdot t$	$-\sigma_0$
4	1	$\sum_{j=0}^{4} g_{0j}\cdot(p-p_0)^j$

Isentropic (or adiabatic) compressibility, κ,

$$\kappa=-\frac{1}{v}\left(\frac{\partial v}{\partial P}\right)_{\sigma}=K-\frac{\alpha^2 T v}{c_P}= \frac{\left(\partial^2 g/\partial T\,\partial P\right)^2-\left(\partial^2 g/\partial T^2\right)_P\left(\partial^2 g/\partial P^2\right)_T}{(\partial g/\partial P)_T\left(\partial^2 g/\partial T^2\right)_P}. \quad (11)$$

Isochoric pressure coefficient, β,

$$\beta=\frac{1}{P}\left(\frac{\partial P}{\partial T}\right)_{v}=-\frac{\partial^2 g/\partial T\,\partial P}{P\cdot\left(\partial^2 g/\partial P^2\right)_T} \quad (12)$$

We finally note that arbitrary mixed partial derivatives of the potential function (Eqs. 1, 2) can be expressed in closed analytical form. The Gibbs function, depending on two independent variables, can be written as a sum of products of functions, each depending on only one variable,

$$g(T,P)=g_U Re\left\{\sum_{k=1}^{4}\Theta_k(T)\,\Phi_k(P)\right\}. \quad (13)$$

The four pairs of functions of Eq. (13) are listed in Table 3.

This factorisation can help in the formal coding of higher, mixed derivatives due to the separation of variables,

$$\frac{\partial^{(n+m)} g(T,P)}{\partial T^n\,\partial P^m}=g_U Re\left\{\sum_{k=1}^{4}\frac{\mathrm{d}^n\Theta_k(T)}{\mathrm{d}T^n}\cdot\frac{\mathrm{d}^m\Phi_k(P)}{\mathrm{d}P^m}\right\}. \quad (14)$$

The nontrivial derivatives of the functions in Table 3 as required for Eq. (14) are in explicit form for k=1 and k=2:

$$\frac{\mathrm{d}\Theta_k}{\mathrm{d}T}=\frac{\tau}{T_t}\left[\ln(t_k+t)-\ln(t_k-t)-\frac{2t}{t_k}\right], \quad (15)$$

$$\frac{\mathrm{d}^2\Theta_k}{\mathrm{d}T^2}=\frac{\tau}{T_t^2}\left[\frac{1}{t+t_k}-\frac{1}{t-t_k}-\frac{2}{t_k}\right], \quad (16)$$

$$\frac{\mathrm{d}^n\Theta_k}{\mathrm{d}T^n}=\tau\cdot(-T_t)^{-n}(n-2)!\cdot\left[(t+t_k)^{(1-n)}-(t-t_k)^{(1-n)}\right], \quad \text{for } n>2 \quad (17)$$

$$\frac{\mathrm{d}^m\Phi_2}{\mathrm{d}P^m}=P_t^{-m}\sum_{j=m}^{2} r_{2j}\frac{j!}{(j-m)!}(p-p_0)^{(j-m)}, \quad (18)$$

$$\frac{\mathrm{d}^m\Phi_4}{\mathrm{d}P^m}=P_t^{-m}\sum_{j=m}^{4} g_{0j}\frac{j!}{(j-m)!}(p-p_0)^{(j-m)}. \quad (19)$$

The mathematical proof is easily obtained by complete induction.

3 Source code

The source code implementations in Visual Basic, Fortran and C++ have been developed independently by these authors, rather than been created as translations of a single template program. The codes contain a common set of same fundamental functions. Additional and auxiliary functions are different, as are the program organisation and language-specific details. For clarity, we provide separate descriptions for all three code samples despite of their various similarities. Implemented are the Gibbs function and its partial derivates up to third order. Independent variables, i.e. input parameters of the functions, are absolute temperature T in K and absolute pressure P in Pa. P includes atmospheric pressure, in opposite to many oceanographic formulae which are based on applied pressure $p=P-101325$ Pa, taken relative to standard pressure at the sea surface. Returned values are measured in appropriate combinations of basic SI units, i.e. kg, J, Pa, m, etc.

3.1 Visual Basic code

The Visual Basic (VB) code comes in five basic files:

- **Gibbs04.vbp**: defines the program components and properties,
- **Gibbs04.frm**: defines the program window and runs the start-up program,
- **Gibbs04.frx**: defines additional properties of the program window,
- **Complex.bas**: implements complex number arithmetic functions,
- **Gibbs04.bas**: contains the thermodynamic functions of ice.

Table 4. Ice properties implemented in VB and their calls.

quantity	function call	unit	eq.
g	ice_g(T, P)	J/kg	2.1
$(\partial g/\partial T)_P$	ice_dgdt(T, P)	J/(kg K)	2.4
$(\partial g/\partial P)_T$	ice_dgdp(T, P)	m^3/kg	2.3
$\left(\partial^2 g/\partial T^2\right)_P$	ice_d2gdt2(T, P)	J/(kg K^2)	2.10
$\left(\partial^2 g/\partial T\partial P\right)$	ice_d2gdtdp(T, P)	m^3/(kg K)	2.9
$\left(\partial^2 g/\partial P^2\right)_T$	ice_d2gdp2(T, P)	m^3/(kg Pa)	2.8
$\left(\partial^3 g/\partial T^3\right)_P$	ice_d3gdt3(T, P)	J/(kg K^3)	
$\left(\partial^3 g/\partial T^2\partial P\right)$	ice_d3gdt2dp(T, P)	m^3/(kg K^2)	
$\left(\partial^3 g/\partial T\partial P^2\right)$	ice_d3gdtdp2(T, P)	m^3/(kg K Pa)	
$\left(\partial^3 g/\partial P^3\right)_T$	ice_d3gdp3(T, P)	m^3/(kg Pa^2)	
ρ	ice_density(T, P)	kg/m^3	2.3
h	ice_enthalpy(T, P)	J/kg	2.6
σ	ice_entropy(T, P)	J/(kg K)	2.4
f	ice_free_energy(T, P)	J/kg	2.5
g	ice_free_enthalpy(T, P)	J/kg	2.1
c_P	ice_heat_capacity(T, P)	J/(kg K)	2.10
e	ice_internal_energy(T, P)	J/kg	2.7
κ	ice_isentropic_compressibility(T, P)	1/Pa	2.11
K	ice_isothermal_compressibility(T, P)	1/Pa	2.8
β	ice_pressure_coefficient(T, P)	1/K	2.12
v	ice_specific_volume(T, P)	m^3/kg	2.3
α	ice_thermal_expansion(T, P)	1/K	2.9

The start-up routine is **Form_Load** in **Gibbs04.frm**. It calls **FW04demo(T_degC, P_dBar)** with the values **T_degC**=−10°C and **P_dBar**=1000 dBar (applied pressure), which produces upon execution the output in the immediate (**Debug**) window of the VB development environment:

T_degC=−10°C
T_abs=263.15 K
P_dBar=1000 dBar
P_abs=10101325 Pa
g −1606.4511744898 J/kg
dg/dT 1299.04800566945 J/(kg K)
dg/dP 1.08787119642917E-03 mˆ3/kg
d2g/dT2 −7.67906117056837 J/(kg Kˆ2)
d2g/dTdP 1.66429674881775E-07 mˆ3/(kg K)
d2g/dP2 −1.25548611251055E-13 mˆ3/(kg Pa)
d3g/dT3 1.41439364919413E-03 J/(kg Kˆ3)
d3g/dT2dP 5.61033076359349E-10 mˆ3/(kg Kˆ2)
d3g/dTdP2 −2.15950418441553E-16 mˆ3/(kg K Pa)
d3g/dP3 8.83364207555687E-23 mˆ3/(kg Paˆ2)
spec. volume 1.08787119642917E-03 mˆ3/kg
density 919.226470268169 kg/mˆ3
enthalpy −343450.933866404 J/kg
entropy −1299.04800566945 J/(kg K)
free energy −12595.3916877597 J/kg
int. energy −354439.874379674 J/kg
heat capacity 2020.74494703507 J/(kg K)
therm. exp. 1.52986562589453E-04 1/K
isoth. compr. 1.15407606767377E-10 1/Pa
adiab. compr. 1.12091901460767E-10 1/Pa
press. coeff. 0.131232229564815 1/K
d(isent. compr)/dP −5.99025933097889E-20 1/Paˆ2

These values may serve as check values whether the implementation works properly.

The thermodynamic functions provided in the module **Gibbs04.bas** contain the functions listed in Table 4, as well as some auxiliary (private) functions not intended for public calls. Prior to any of these function calls, **Sub InitG_FW04** must be executed to initialise the field of coefficients.

3.2 FORTRAN code

The Fortran code comes in two basic files:

- **main.for**: runs the main program
- **ice-lib.for**: contains the thermodynamic functions of ice

The start-up routine is **Program Ice**. It runs a computation of ice properties with the values **T_abs**=263.15 K and

Table 5. Ice properties implemented in Fortran and their calls.

quantity	function call	unit	Eq.
g	ice_g(T, P)	J/kg	2.1
$(\partial g/\partial T)_P$	ice_dgdt(T, P)	J/(kg K)	2.4
$(\partial g/\partial P)_T$	ice_dgdp(T, P)	m^3/kg	2.3
$\left(\partial^2 g/\partial T^2\right)_P$	ice_d2gdt2(T, P)	J/(kg K^2)	2.10
$\left(\partial^2 g/\partial T\partial P\right)$	ice_d2gdtdp(T, P)	m^3/(kg K)	2.9
$\left(\partial^2 g/\partial P^2\right)_T$	ice_d2gdp2(T, P)	m^3/(kg Pa)	2.8
$\left(\partial^3 g/\partial T^3\right)_P$	ice_d3gdt3(T, P)	J/(kg K^3)	
$\left(\partial^3 g/\partial T^2\partial P\right)$	ice_d3gdt2dp(T, P)	m^3/(kg K^2)	
$\left(\partial^3 g/\partial T\partial P^2\right)$	ice_d3gdtdp2(T, P)	m^3/(kg K Pa)	
$\left(\partial^3 g/\partial P^3\right)_T$	ice_d3gdp3(T, P)	m^3/(kg Pa^2)	
ρ	ice_density(T, P)	kg/m^3	2.3
h	ice_enthalpy(T, P)	J/kg	2.6
σ	ice_entropy(T, P)	J/(kg K)	2.4
f	ice_free_energy(T, P)	J/kg	2.5
g	ice_free_enthalpy(T, P)	J/kg	2.1
c_P	ice_heat_capacity(T, P)	J/(kg K)	2.10
e	ice_internal_energy(T, P)	J/kg	2.7
κ	ice_isentropic_compressibility(T, P)	1/Pa	2.11
$(\partial\kappa/\partial P)_T$	ice_disentropic_compressibilitydp(T,P)	1/Pa^2	
K	ice_isothermal_compressibility(T, P)	1/Pa	2.8
β	ice_pressure_coefficient(T, P)	1/K	2.12
v	ice_specific_volume(T, P)	m^3/kg	2.3
α	ice_thermal_expansion(T, P)	1/K	2.9

P_abs=10101325 Pa, which produces upon execution the output in the default window:

```
T_abs 263.150000000000 K
P_abs 10101325.0000000 Pa
g −1606.45117448973 J/kg
dg/dT 1299.04800566944 J/(kg K)
dg/dP 1.087871196429174E-003 m**3/kg
d2g/dT2 −7.67906117056837 J/(kg K**2)
d2g/dTdP 1.664296748817753E-007 m**3/(kg K)
d2g/dP2 −1.255486112510547E-013 m**3/(kg Pa)
d3g/dT3 1.414393649194133E-003 J/(kg K**3)
d3g/dT2dP 5.610330763593495E-010 m**3/(kg K**2)
d3g/dTdP2 −2.159504184415537E-016 m**3/(kg Pa K)
d3g/dP3 8.833642075556870E-023 m**3/(kg Pa**2)
specific volume 1.087871196429174E-003 m**3/kg
density 919.226470268169 kg/m**3
enthalpy −343450.933866404 J/kg
entropy −1299.04800566944 J/(kg K)
free energy −12595.3916877597 J/kg
internal energy −354439.874379674 J/kg
heat capacity 2020.74494703507 J/(kg K)
therm. expansion 1.529865625894533E-004 1/K
isoth. compr. 1.154076067673776E-010 1/Pa
isentr. compr. 1.120919014607667E-010 1/Pa
press. coefficient 0.131232229564815 1/K
disentr. compr./dP −5.990259330978887E-020 1/Pa**2
```

These values may serve as check values whether the implementation works properly.

The thermodynamic functions provided in the module **icelib.for** contain the functions listed in Table 5, as well as some auxiliary functions not intended for public calls. Prior to any of these function calls, **SUBROUTINE DAT** must be executed to initialise the field of coefficients.

3.3 C++ code

The C++ code comes in five basic files:

- **main.cpp**: runs the main program,
- **IceIh_FW.cpp**: contains the thermodynamic functions of ice,
- **IceIh_FW_Aux.h**: header file for **IceIh_FW.cpp**,
- **IceIh_FW_Aux.cpp**: contains numerical constants and auxiliary functions,

Table 6. Ice properties implemented in C++ and their calls.

quantity	function call	unit	Eq.
g	ice_g(T, P)	J/kg	2.1
$(\partial g/\partial T)_P$	ice_dgdt(T, P)	J/(kg K)	2.4
$(\partial g/\partial P)_T$	ice_dgdp(T, P)	m^3/kg	2.3
$\left(\partial^2 g/\partial T^2\right)_P$	ice_d2gdt2(T, P)	J/(kg K^2)	2.10
$\left(\partial^2 g/\partial T \partial P\right)$	ice_d2gdtdp(T, P)	m^3/(kg K)	2.9
$\left(\partial^2 g/\partial P^2\right)_T$	ice_d2gdp2(T, P)	m^3/(kg Pa)	2.8
$\left(\partial^3 g/\partial T^2 \partial P\right)$	ice_d3gdt2dp(T, P)	m^3/(kg K^2)	
$\left(\partial^3 g/\partial T \partial P^2\right)$	ice_d3gdtdp2(T, P)	m^3/(kg K Pa)	
$\left(\partial^3 g/\partial P^3\right)_T$	ice_d3gdp3(T, P)	m^3/(kg Pa^2)	
ρ	ice_density(T, P)	kg/m^3	2.3
h	ice_enthalpy(T, P)	J/kg	2.6
σ	ice_entropy(T, P)	J/(kg K)	2.4
f	ice_free_energy(T, P)	J/kg	2.5
c_P	ice_heat_capacity(T, P)	J/(kg K)	2.10
e	ice_internal_energy(T, P)	J/kg	2.7
κ	ice_isentropic_compressibility(T, P)	1/Pa	2.11
$(\partial \kappa/\partial P)_T$	ice_disentropic_compressibilitydp(T,P)	1/Pa^2	
K	ice_isothermal_compressibility(T, P)	1/Pa	2.8
β	ice_pressure_coefficient(T, P)	Pa/K	2.12
v	ice_specific_volume(T, P)	m^3/kg	2.3
α	ice_thermal_expansion(T, P)	1/K	2.9

– **IceIh_FW.h**: header file for, **IceIh_FW_Aux.cpp**.

The ANSI/ISO C++ standard (Schildt, 1998) implements a **complex** class which represents complex numbers and defines a series of functions that operate on objects of type **complex**, including a logarithmic function; there is no need to provide user-defined operations on complex numbers.

The C++ code was compiled in the Windows and in the LINUX environments. Borland C++ Builder 5.0 (Windows) and GNU g++ 3.3.1 (LINUX) compilers were used to generate the executable files.

The start-up routine is **main**. It runs a computation of ice properties with the values.

T_abs=263.15 K and **P_abs**=10101325 Pa, which produces upon execution the output in the default DOS window (Windows environment) or in the default terminal window (LINUX environment).

The Windows output:

T_abs 263.15 K
P_abs 10101325 Pa
g −1606.45117448966 J/kg
dgdT 1299.04800566944 J/(kg K)
dgdP 0.00108787119642917 mˆ3/kg
d2gdT2 −7.67906117056837 J/(kg Kˆ2)
d2gdTdP 1.66429674881775e-07 mˆ3/(kg K)
d2gdP2 −1.25548611251055e-13 mˆ3/(kg Pa)
d3gdT2dP 5.61033076359349e-10 mˆ3/(kg Kˆ2)
d3gdTdP2 −2.15950418441554e-16 mˆ3/(kg K Pa)
d3gdP3 8.83364207555687e-23 mˆ3/(kg Paˆ2)
spec. volume 0.00108787119642917 mˆ3/kg
density 919.226470268169 kg/mˆ3
enthalpy −343450.933866404 J/kg
entropy −1299.04800566944 J/(kg K)
free energy −12595.3916877596 J/kg
int. energy −354439.874379674 J/kg
heat capacity 2020.74494703507 J/(kg K)
therm. exp. 0.000152986562589453 1/K
isoth. compr. 1.15407606767378e-10 1/Pa
isent. compr. 1.12091901460767e-10 1/Pa
press. coeff. 0.131232229564815 1/K
d(isent. compr)/dP −5.99025933097888e-20 1/Paˆ2

The LINUX output:

T_abs 263.15 K
P_abs 10101325 Pa
g −1606.45117448957 J/kg
dgdT 1299.04800566944 J/(kg K)
dgdP 0.00108787119642917 mˆ3/kg
d2gdT2 −7.67906117056837 J/(kg Kˆ2)
d2gdTdP 1.66429674881775e-07 mˆ3/(kg K)
d2gdP2 −1.25548611251055e-13 mˆ3/(kg Pa)

Table 7. Freezing temperatures in °C of seawater with practical salinities S from 0 to 40 psu under applied pressure p (relative to normal pressure P_0) from 0 to 100 MPa (0 to 10 000 dbar). Values computed by Eq. (22).

p/MPa	0 psu	5 psu	10 psu	15 psu	20 psu	25 psu	30 psu	35 psu	40 psu
0	+0.003	−0.272	−0.54	−0.809	−1.081	−1.356	−1.636	−1.921	−2.208
10	−0.756	−1.031	−1.3	−1.569	−1.842	−2.118	−2.4	−2.685	−2.974
20	−1.546	−1.822	−2.091	−2.361	−2.634	−2.912	−3.194	−3.481	−3.772
30	−2.367	−2.643	−2.913	−3.184	−3.458	−3.736	−4.02	−4.308	−4.6
40	−3.219	−3.495	−3.766	−4.037	−4.312	−4.592	−4.877	−5.167	−5.46
50	−4.1	−4.377	−4.649	−4.921	−5.197	−5.478	−5.764	−6.055	−6.35
60	−5.012	−5.289	−5.562	−5.835	−6.112	−6.394	−6.682	−6.975	−7.271
70	−5.953	−6.231	−6.505	−6.779	−7.058	−7.341	−7.63	−7.924	−8.223
80	−6.924	−7.203	−7.478	−7.753	−8.033	−8.318	−8.608	−8.904	−9.204
90	−7.924	−8.204	−8.48	−8.757	−9.038	−9.324	−9.617	−9.915	−10.217
100	−8.954	−9.235	−9.512	−9.791	−10.073	−10.361	−10.656	−10.956	−11.26

d3gdT2dP 5.61033076359349e-10 mˆ3/(kg Kˆ2)
d3gdTdP2 −2.15950418441554e-16 mˆ3/(kg K Pa)
d3gdP3 8.83364207555687e-23 mˆ3/(kg Paˆ2)
spec. volume 0.00108787119642917 mˆ3/kg
density 919.226470268169 kg/mˆ3
enthalpy −343450.933866404 J/kg
entropy −1299.04800566944 J/(kg K)
free energy −12595.3916877595 J/kg
int. energy −354439.874379674 J/kg
heat capacity 2020.74494703507 J/(kg K)
therm. exp. 0.000152986562589453 1/K
isoth. compr. 1.15407606767378e-10 1/Pa
isent. compr. 1.12091901460767e-10 1/Pa
press. coeff. 0.131232229564815 1/K
d(isent. compr)/dP −5.99025933097889e-20 1/Paˆ2

The values above may serve to check whether the implementation works properly. Note slight differences in the values of g, free energy, and d(isent. compr)/dP produced by the Windows and the LINUX executables.

The thermodynamic functions provided in the module **IceIh_FW.cpp** contain the functions listed in Table 6. Auxiliary (private) functions not intended for public calls are contained in the module **IceIh_FW_Aux.cpp**. All of the functions listed in the Table 6 are embedded in the namespace **IceIh_FW** which protects them against possible naming conflicts.

4 Aqueous phase equilibria

Equilibria between ice and liquid water or water vapour require equal chemical potentials of water of the coexisting phases, which are available from the IAPWS-95 free enthalpy of liquid pure water, $g^W(T, P)$, and of water vapour, $g^V(T, P)$ (Wagner and Pruß, 2002).

Melting point temperature of ice, $T_M(P)$, is implicitly given by

$$g(T_M, P) = g^W(T_M, P). \tag{20}$$

Vapour pressure, $P_{vap}(T, P)$, over ice at pressure P, is implicitly given by

$$g(T, P) = g^V\left(T, P_{vap}\right). \tag{21}$$

Equilibria between ice and seawater (i.e. sea ice) require equal chemical potentials of water of those phases, which are available from the free enthalpy of seawater, $g^{SW}(S, T, P)$ (Feistel, 2003, 2004). Here, S is the practical salinity of the liquid phase (i.e. seawater).

Freezing point temperature of seawater, $T_F(S, P)$, is implicitly given by

$$g(T_F, P) = g^{SW}(S, T_F, P) - S\left(\frac{\partial g^{SW}}{\partial S}\right)_{P,T=T_F}. \tag{22}$$

Freezing points of air-free seawater in the oceanic range of salinities and pressures are listed in Table 7; their accuracy is estimated as 2 mK at atmospheric pressure and 30 mK at high pressures (Feistel and Wagner, 2005b). The additional freezing point lowering due to saturation with air is in the range of 0–3 mK (Feistel, 2003; Jackett et al., 2005). Precipitation of sodium sulfate out of seawater starts at −8.2°C at atmospheric pressure, or at brine salinities of about 150 psu (Schwertfeger, 1963), but perhaps even at −6.3°C already (Marion et al., 1999), corresponding to about 115 psu, and is likely not relevant for the values given in Table 7, even though its pressure dependence is not known. The low solubility of calcium carbonate, however, hypothetically causing precipitation already below −2.2 °C (Marion, 2001) or at about 40 psu, if confirmed may well exhibit some quantitative influence, however. Calcium contributes to standard sea salt with about 1.2 mass percent (Feistel, 2003). Measurements of freezing, high-salinity Mediterranean seawater

down to −14°C were reported by Herut et al. (1990). These data may need to be considered in future extensions to higher salinities of the Gibbs function of seawater but do not concern the Gibbs function of ice presented in this paper.

For vanishing salinity, S=0, this Eq. (22) can be used instead of Eq. (20).

Brine salinity of sea ice, $S_B(T, P)$, is implicitly given by

$$g(T, P) = g^{SW}(S_B, T, P) - S_B \left(\frac{\partial g^{SW}}{\partial S}\right)_{T,P,S=S_B} . \quad (23)$$

Given the bulk salinity s of sea ice, which equals its practical salinity after complete melting, the (liquid) mass fraction of brine, w, is given by,

$$w\,(s, T, P) = \frac{s}{S_B\,(T, P)} . \quad (24)$$

Free enthalpy of sea ice, g^{SI}, as a function of bulk salinity, temperature and pressure, is obtained as (Feistel and Hagen, 1998)

$$g^{SI}\,(s, T, P) = (1 - w\,(s, T, P)) \cdot g\,(T, P) + w\,(s, T, P) \cdot g^{SW}\,(S_B\,(T, P)\,, T, P)\,. \quad (25)$$

All properties of sea ice with constant bulk salinity can be computed from this function in analogy to Eqs. (1)–(12) by substituting g by g^{SI}. Some important formulae are the following ones (Feistel and Hagen, 1998):

Specific volume, v^{SI}, and density, ρ^{SI}=$1/v^{SI}$, of sea ice is given by

$$v^{SI} = \left(\frac{\partial g^{SI}}{\partial P}\right)_{s,T} = (1 - w) \cdot v + w \cdot v^{SW}\,(S_B, T, P) \quad (26)$$

with $v^{SW}\,(S, T, P) = \left(\frac{\partial g^{SW}}{\partial P}\right)_{T,S}$.

Specific enthalpy, h^{SI}, of sea ice is given by

$$h^{SI} = g^{SI} - T \left(\frac{\partial g^{SI}}{\partial T}\right)_{s,P} = (1 - w) \cdot h + w \cdot h^{SW}\,(S_B, T, P) \quad (27)$$

with

$$h^{SW}\,(S, T, P) = g^{SW}\,(S, T, P) - T \left(\frac{\partial g^{SW}}{\partial T}\right)_{P,S} .$$

Specific heat capacity, c_P^{SI}, of sea ice is given by

$$c_P^{SI} = \left(\frac{\partial h^{SI}}{\partial T}\right)_{s,P} = (1-w) \cdot c_P + w \cdot \left[c_P^{SW}\,(S_B, T, P) + c_P^{Melt}\,(S_B, T, P)\right] \quad (28)$$

with

$$c_P^{SW}\,(S, T, P) = \left(\frac{\partial h^{SW}}{\partial T}\right)_{P,S}$$

and

$$c_P^{Melt}\,(S, T, P) = \frac{\left[\left(h^{SW} - h\right)/S - \left(\partial h^{SW}/\partial S\right)_{T,P}\right]^2}{T \cdot \left(\partial^2 g^{SW}/\partial S^2\right)_{T,P}} .$$

Isothermal compressibility, K^{SI}, of sea ice is given by

$$v^{SI} K^{SI} = -\left(\frac{\partial v^{SI}}{\partial P}\right)_{s,T} = (1 - w) \cdot v \cdot K + w \cdot v^{SW} \cdot \left[K^{SW}\,(S_B, T, P) + K^{Melt}\,(S_B, T, P)\right] \quad (29)$$

with

$$K^{SW}\,(S, T, P) = -\frac{1}{v^{SW}} \left(\frac{\partial v^{SW}}{\partial P}\right)_{T,S}$$

and

$$K^{Melt}\,(S, T, P) = \frac{1}{v^{SW}} \frac{\left[\left(v^{SW} - v\right)/S - \left(\partial v^{SW}/\partial S\right)_{T,P}\right]^2}{\left(\partial^2 g^{SW}/\partial S^2\right)_{T,P}} .$$

Cubic thermal expansion coefficient, α^{SI}, of sea ice is given by

$$v^{SI} \alpha^{SI} = \left(\frac{\partial v^{SI}}{\partial T}\right)_{s,P} = (1 - w) \cdot v \cdot \alpha + w \cdot v^{SW} \cdot \left[\alpha^{SW}\,(S_B, T, P) + \alpha^{Melt}\,(S_B, T, P)\right] \quad (30)$$

with

$$\alpha^{SW}\,(S, T, P) = \frac{1}{v^{SW}} \left(\frac{\partial v^{SW}}{\partial T}\right)_{P,S} .$$

and

$$\alpha^{Melt}\,(S, T, P) = \frac{1}{v^{SW}} \frac{\left[\left(v^{SW} - v\right)/S - \left(\partial v^{SW}/\partial S\right)_{T,P}\right]\left[\left(h^{SW} - h\right)/S - \left(\partial h^{SW}/\partial S\right)_{T,P}\right]}{T \cdot \left(\partial^2 g^{SW}/\partial S^2\right)_{T,P}} .$$

The "melting" contributions to heat capacity, c_P^{Melt}, compressibility, K^{Melt}, and thermal expansion, α^{Melt}, consist of the latent parts due to the partial transition process between ice and water, as well as effects due to varying brine salinity, haline contraction and dilution heat. They are numerically very important and may even exceed the single contributions of both "pure" phases, ice and brine.

Edited by: C. Döös

References

Alberty, R. A.: Use of Legendre transforms in chemical thermodynamics, Pure Appl. Chem., 73, 1349–1380, 2001.

Feistel, R: Equilibrium thermodynamics of seawater revisited, Progr. Oceanogr., 31, 101–179, 1993.

Feistel, R: A new extended Gibbs thermodynamic potential of seawater, Progr. Oceanogr., 58, 43-115, 2003, Corrigendum 61, 99, 2004.

Feistel, R: Numerical implementation and oceanographic application of the Gibbs thermodynamic potential of seawater, Ocean Sci., 1, 9–16, 2005, **SRef-ID: 1812-0792/os/2005-1-9**.

Feistel, R. and Hagen, E.: On the GIBBS thermodynamic potential of seawater, Progr. Oceanogr., 36, 249–327, 1995.

Feistel, R. and Hagen, E.: A Gibbs thermodynamic potential of sea ice, Cold Reg. Sci. Technol., 28, 83–142, 1998, Corrigendum 29, 173–176, 1999.

Feistel, R. and Wagner, W.: A comprehensive Gibbs thermodynamic potential of ice, in: Water, Steam, and Aqueous Solutions for Electric Power, edited by: Nakahara, M., Matubayashi, N., Ueno, M., Yasuoka, K., Watanabe, K., Proc. 14th Int. Conf. Prop. Water Steam (ICPWS), Kyoto, Japan, Maruzen Co. Ltd., 751–756, 2005a.

Feistel, R. and Wagner, W.: High-pressure thermodynamic Gibbs functions of ice and sea ice, J. Mar. Res., 63, 95–139, 2005b.

Hagen, E. and Feistel, R.: Climatic Turning Points and Regime Shifts in the Baltic Sea Region: The Baltic Winter Index (1659–2002), Boreal Env. Res., 10, 211–224, 2005.

Herut, B., Starinsky, A., Katz, A., and Bein, A.: The role of seawater freezing in the formation of subsurface brines, Geochim. Cosmochim. Acta, 54, 13–21, 1990.

Jackett, D. R., McDougall, T. J., Feistel, R., Wright, D. G., and Griffies, S. M.: Updated algorithms for density, potential temperature, conservative temperature and freezing temperature of seawater, J. Atm. Oceanic Technol., in press, 2005.

Jones, P. D.: Hemispheric surface air temperature variations: a reanalysis and update to 1993, J. Climate, 7, 1794–1802, 1994.

Marion, G. M. and Farren, R. E.: Mineral solubilities in the Na-K-Mg-Ca-Cl-SO_4-H_2O system: A re-evaluation of the sulphate chemistry in the Spencer-Møller-Weare model, Geochim. Cosmochim. Acta, 63, 1305–1318, 1999.

Marion, G. M.: Carbonate mineral solubility at low temperatures in the Na-K-Mg-Ca-H-Cl-SO_4-OH-HCO_3-CO_3-CO_2-H_2O system, Geochim. Cosmochim. Acta, 65, 1883–1896, 2001.

Millero, F. J.: Freezing point of seawater, Unesco techn. pap. mar. sci., 28, 29–35, 1978.

Schildt, H.: C++: The complete reference, third edition, Osborne McGraw-Hill, Berkeley, 1998.

Schwertfeger, P.: The thermal properties of sea ice, J. Glaciol., 4, 789–807, 1963.

Span, R. and Wagner, W.: Equations of state for technical applications: I. Simultaneously optimized functional forms for nonpolar and polar fluids, Int. J. Thermophys., 24, 1–39, 2003.

Tillner-Roth, R.: Fundamental equations of state, Shaker Verlag, Aachen, 1998.

Wagner, W. and Pruß, A.: The IAPWS formulation 1995 for the thermodynamic properties of ordinary water substance for general and scientific use, J. Phys. Chem. Ref. Data, 31, 387–535, 2002.

Wagner, W., Saul, A. and Pruß, A.: International equations for the pressure along the melting and along the sublimation curve of ordinary water substance, J. Phys. Chem. Ref. Data, 23, 515–527, 1994.